AMENDMENTS TO THE CLAIMS

1. (Canceled)

2. (Canceled)

3. (Currently Amended) ~~The processor of claim 2 further including~~ A processor comprising:

an apparatus to rotate registers in software pipelined loops;

a register rotation prediction unit to predict register addresses for future loop iterations;

a buffer to hold buffered instructions with predicted register addresses; and

unarchitected predicate registers to predicate the buffered instructions.

4. (Currently Amended) ~~The processor of claim 2 wherein~~ A processor comprising:

an apparatus to rotate registers in software pipelined loops;

a register rotation prediction unit to predict register addresses for future loop iterations;

a buffer to hold buffered instructions with predicted register addresses; and

the predicted register addresses are such that the buffered instructions can be issued simultaneously with a branch instruction.

5. (Currently Amended) ~~The processor of claim 1 further including~~ A processor comprising:

an apparatus to rotate registers in software pipelined loops;

a register rotation prediction unit to predict register addresses for future loop iterations; and

a hint register to encode prediction hints for the register rotation prediction unit.

6. (Original) The processor of claim 5 wherein the hint register is configured to hold static hints generated by a compiler.

7. (Original) The processor of claim 5 wherein the hint register is configured to hold dynamic hints generated at runtime.

8. (Original) The processor of claim 5 wherein the hint register includes a field to specify an iteration distance.

9. (Currently Amended) ~~The processor of claim 1 further comprising~~ A processor comprising:

an apparatus to rotate registers in software pipelined loops;

a register rotation prediction unit to predict register addresses for future loop iterations;

a plurality of unarchitected frame marker registers.

10. (Original) The processor of claim 9 wherein the register rotation prediction unit comprises speculation decision making hardware to compute values for the plurality of unarchitected frame marker registers.

11. (Original) The processor of claim 10 further comprising register renaming hardware in a pipeline, the register renaming hardware being responsive to the plurality of unarchitected frame marker registers.

12. (Currently Amended) The processor of claim 5 [[1]] further comprising a trace cache.

13. (Original) The processor of claim 12 wherein the trace is configured to hold a prediction hint for each trace.

14. (Original) The processor of claim 13 further comprising a trace cache fill unit to apply register rotation prediction to traces as traces are constructed.

15. (Canceled)

16. (Canceled)

17. (Currently Amended) ~~The processing system of claim 16 further comprising:~~ A processing system comprising:

an execution pipeline;

cache memory coupled to the execution pipeline to hold processor instructions arranged in a software loop;

register rotation prediction hardware to predict physical register values for the processor instructions in future iterations of the software loop;

a software pipeline instruction buffer coupled between the execution pipeline and the register rotation prediction hardware to hold the processor instructions in future iterations of the software loop; and

at least one unarchitected frame marker register coupled to the register rotation prediction hardware to hold predicted register offsets for future iterations.

18. (Original) The processing system of claim 17 wherein the execution pipeline includes register renaming logic responsive to the at least one unarchitected frame marker register.

19. (Currently Amended) ~~The processing system of claim 16 wherein~~ A processing system comprising:

an execution pipeline;

cache memory coupled to the execution pipeline to hold processor instructions arranged in a software loop;

register rotation prediction hardware to predict physical register values for the processor instructions in future iterations of the software loop;

a software pipeline instruction buffer coupled between the execution pipeline and the register rotation prediction hardware to hold the processor instructions in future iterations of the software loop; and

the register rotation prediction hardware includes a circuit to specify complete physical register addresses for the processor instructions in future iterations of the software loop.

20. (Original) The processing system of claim 19 wherein processor instructions held in the software pipeline instruction buffer include fully specified physical register addresses.

21. (Currently Amended) ~~The processing system of claim 16 wherein~~ A processing system comprising:

an execution pipeline;

cache memory coupled to the execution pipeline to hold processor instructions arranged in a software loop;

register rotation prediction hardware to predict physical register values for the processor instructions in future iterations of the software loop;

a software pipeline instruction buffer coupled between the execution pipeline and the register rotation prediction hardware to hold the processor instructions in future iterations of the software loop; and

the execution pipeline is configured to speculatively execute instructions received from the software pipeline instruction buffer.

22. (Original) The processing system of claim 21 further comprising a plurality of unarchitected predicate registers, wherein the instructions within the software pipeline instruction buffer are predicated on at least one of the plurality of unarchitected predicate registers.

23. (Canceled)

24. (Canceled)

25. (Currently Amended) ~~The method of claim 24~~ wherein A method of executing a software pipelined loop comprising:

rotating registers for each iteration of the loop;

predicting register rotations for future iterations of the loop;

the software pipelined loop comprises at least one branch instruction, the method further comprising issuing at least one non-branch instruction simultaneously with the at least one branch instruction; and

the at least one non-branch instruction is predicated on an unarchitected predicate register.

26. (Currently Amended) ~~The method of claim 24 further comprising~~ A method of executing a software pipelined loop comprising:

rotating registers for each iteration of the loop;

predicting register rotations for future iterations of the loop;

the software pipelined loop comprises at least one branch instruction, the method further comprising issuing at least one non-branch instruction simultaneously with the at least one branch instruction; and

speculatively removing stop bits from the at least one branch instruction.

27. (Original) The method of claim 26 further comprising speculatively executing the at least one non-branch instruction.

28. (Currently Amended) ~~The method of claim 23 wherein predicting comprises:~~ A method of executing a software pipelined loop comprising:

rotating registers for each iteration of the loop;

predicting register rotations for future iterations of the loop;

responsive to a hint register, predicting register rotations for more than one iteration in the future; and

modifying at least one unarchitected frame marker register.

29. (Original) The method of claim 28 further comprising:
speculatively executing instructions for the more than one iteration in the future; and
squashing the speculative execution if a data dependence is violated.

30. (Original) The method of claim 29 further comprising modifying the hint register when speculative execution is squashed.